

## Web

Results 1 - 10 of about 93,100 for theorem prover cycles budget. (0.20 seconds)**[PDF] SRC 171**File Format: PDF/Adobe Acrobat - [View as HTML](#)

and submit the conjecture to an appropriate automatic **theorem prover**. ... within the **cycle budget**. Finally, the satisfiability solver finds a solution or ... [www.hpl.hp.com/techreports/Compaq-DEC/SRC-RR-171.pdf](http://www.hpl.hp.com/techreports/Compaq-DEC/SRC-RR-171.pdf) - [Similar pages](#)

**[PS] 1 The Straight-Line Automatic Programming Problem** Rajeev Joshi ...File Format: Adobe PostScript - [View as Text](#)In other words, we apply automatic **theorem-proving** methods to the problem of ...That is, the code performs each goal update within the **cycle budget**. ...[www.hpl.hp.com/techreports/2003/HPL-2003-236.ps](http://www.hpl.hp.com/techreports/2003/HPL-2003-236.ps) - [Similar pages](#)[ More results from [www.hpl.hp.com](http://www.hpl.hp.com) ]**[PS] Denali: A Goal-directed Superoptimizer** Rajeev Joshi Greg Nelson ...File Format: Adobe PostScript - [View as Text](#)The search principle: A refutation-based automatic **theorem prover** is in fact a... For a fixed E-graph and a fixed **cycle budget**, the constraints are ...[theory.lcs.mit.edu/~randall/papers/denali.ps](http://theory.lcs.mit.edu/~randall/papers/denali.ps) - [Similar pages](#)**[PS] 9 August 2001 SRC Research Report 171 Denali: A Goal-directed ...**File Format: Adobe PostScript - [View as Text](#)and submit the conjecture to an appropriate automatic **theorem prover**. ... For afixed **cycle budget**, the constraint generator formulates as a boolean ...[ftp.digital.com/pub/Digital/SRC/research-reports/SRC-171.ps.Z](http://ftp.digital.com/pub/Digital/SRC/research-reports/SRC-171.ps.Z) - [Similar pages](#)**[PDF] SRC 171**File Format: PDF/Adobe Acrobat - [View as HTML](#)

and submit the conjecture to an appropriate automatic **theorem prover**. If ... expressions in the original GMA can be computed within the **cycle budget** us- ...

[ftp.digital.com/pub/Digital/SRC/research-reports/SRC-171.pdf](http://ftp.digital.com/pub/Digital/SRC/research-reports/SRC-171.pdf) - [Similar pages](#)**[PS] 30 July 2001 SRC Research Report 171 Denali: a goal-directed ...**File Format: Adobe PostScript - [View as Text](#)Because auto-matic **theorem-provers** treat entire arrays as values, the update to... For a fixed E-graph and a fixed **cycle budget**, the constraints are ...[gatekeeper.dec.com/pub/DEC/SRC/research-reports/SRC-171.ps.Z](http://gatekeeper.dec.com/pub/DEC/SRC/research-reports/SRC-171.ps.Z) - [Similar pages](#)**[PDF] Denali: A Goal-directed Superoptimizer**File Format: PDF/Adobe Acrobat - [View as HTML](#)

**theorem prover**. If the proof succeeds, then 8 **cycles** are not enough, and we try again, with, ... For a fixed E-graph and a fixed **cycle budget**, the con- ...

[www.cs.utah.edu/~wilson/compilers/old/papers/pldi02-joshi.pdf](http://www.cs.utah.edu/~wilson/compilers/old/papers/pldi02-joshi.pdf) - [Similar pages](#)**Academic Programs :: Graduate Programs :: Course Descriptions**... including: heuristic problem-solving search and **theorem-proving** techniques,rule-based systems ... The **budget/fiscal year cycle**. Audits. Cost control. ...[www.umtweb.edu/AcademicPrograms/GraduatePrograms/CourseDescriptions.htm](http://www.umtweb.edu/AcademicPrograms/GraduatePrograms/CourseDescriptions.htm) - 82k - [Cached](#) - [Similar pages](#)**Bugzilla Query Page Help**

bitnotes, **Theorem prover** for mathematics applying Shults' ideas. Written completely in Java. **budget**, A console-based **budget** program written in Java. ...

[www.mathcs.bethel.edu/bugzilla/queryhelp.cgi](http://www.mathcs.bethel.edu/bugzilla/queryhelp.cgi) - 63k - Oct 3, 2005 - [Cached](#) - [Similar pages](#)**The Chronicle: Career Network: 05/31/2002**Since our recruiting **cycle** doesn't really end, but just lurches forward from one... to have solved Fermat's next-to-last **theorem**, **proving** that light really ...[chronicle.com/jobs/2002/05/2002053101c.htm](http://chronicle.com/jobs/2002/05/2002053101c.htm) - 34k - Oct 4, 2005 - [Cached](#) - [Similar pages](#)


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

theorem prover cycles budget

USPTO

THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)
Terms used **theorem prover cycles budget**

Found 10,609 of 161,645

Sort results by

relevance

[Save results to a Binder](#)Try an [Advanced Search](#)Try this search in [The ACM Guide](#)

Display results

expanded form

[Search Tips](#)

Open results in a new window

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐**1 [Denali: a goal-directed superoptimizer](#)**

Rajeev Joshi, Greg Nelson, Keith Randall

May 2002

**ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2002 Conference on****Programming language design and implementation**, Volume 37 Issue 5Full text available: [pdf\(179.77 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

This paper provides a preliminary report on a new research project that aims to construct a code generator that uses an automatic theorem prover to produce very high-quality (in fact, nearly mathematically optimal) machine code for modern architectures. The code generator is not intended for use in an ordinary compiler, but is intended to be used for inner loops and critical subroutines in those cases where peak performance is required, no available compiler generates adequately efficient code, ...

**Keywords:** optimizing compiler, superoptimizer**2 [Consistency management in a project management assistant](#)**

Xiaolei Qian, Richard Jullig, Marilyn Daum

October 1990

**ACM SIGSOFT Software Engineering Notes , Proceedings of the fourth ACM SIGSOFT symposium on Software development environments**, Volume 15 Issue 6Full text available: [pdf\(1.15 MB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Object management systems have been identified as the core of object-oriented software development environments. One of the most important objectives of object management systems is to maintain consistency between the vast amount of interrelated objects, which is generated, accessed, and manipulated throughout the software life cycle. Consistency management in such systems is beyond the reach of conventional database technology due to the complex structure and the incompleteness of data, th ...

**3 [AFFIRM summary](#)**

Susan L. Gerhart

July 1981

**ACM SIGSOFT Software Engineering Notes**, Volume 6 Issue 3Full text available: [pdf\(684.52 KB\)](#)Additional Information: [full citation](#), [references](#)**4 [Enhancing cleanroom techniques with refinement calculus](#)**

Michael R. Donat

November 1995

**Proceedings of the 1995 conference of the Centre for Advanced Studies on Collaborative research**Full text available: [pdf\(152.85 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

A refinement calculus is a mathematical technique that provides a mechanism for rigorously transforming specifications into executable programs in a step-wise manner. A by-product of this technique is a formal proof of correctness of both the implementation and the design. These step-wise and rigorous aspects make this an attractive method for reducing software development costs and mitigating risk. The cleanroom technique has been quite successful as a development method [4]. However, it does no ...

**5 [On network design problems: fixed cost flows and the covering steiner problem](#)**

Guy Even, Guy Kortsarz, Wolfgang Slany

July 2005

**ACM Transactions on Algorithms (TALG)**, Volume 1 Issue 1Full text available: [pdf\(277.45 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Network design problems, such as generalizations of the Steiner Tree Problem, can be cast as edge-cost-flow problems. An edge-cost flow problem is a min-cost flow problem in which the cost of the flow equals the sum of the costs of the edges carrying positive flow. We prove a hardness result for the Minimum Edge Cost Flow Problem (MECF). Using the one-round two-prover scenario, we prove that MECF does not admit a  $2^{\log^{1-\epsilon} n}$ -ratio approximation, for every constant ...

**Keywords:** Optimization, approximation, flow, graphs, theory

## 6 Modeling methodology b: Simulation and verification I: from simulation to verification (and back)

Harald Rueß, Leonardo de Moura

December 2003 **Proceedings of the 35th conference on Winter simulation: driving innovation**

Full text available:  [pdf\(198.00 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)

Symbolic evaluation is the execution of software and software designs on inputs given as symbolic or explicit constants along with constraints on these inputs. Efficient symbolic evaluation is now feasible due to recent advances in efficient decision procedures and symbolic model checking. Symbolic evaluation can be applied to partially implemented descriptions and provides wider coverage and greater assurance than testing and traditional simulation alone. Unlike full formal verification, sym ...

## 7 Tools and transformations—rigorous and otherwise—for practical database design

Annon Rosenthal, David Reiner

June 1994 **ACM Transactions on Database Systems (TODS)**, Volume 19 Issue 2

Full text available:  [pdf\(3.19 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

We describe the tools and theory of a comprehensive system for database design, and show how they work together to support multiple conceptual and logical design processes. The Database Design and Evaluation Workbench (DDEW) system uses a rigorous, information-content-preserving approach to schema transformation, but combines it with heuristics, guess work, and user interactions. The main contribution lies in illustrating how theory was adapted to a practical system, and how the consistency ...

**Keywords:** applications of database theory, computer-aided software engineering, data model translation, database design, database equivalence, design heuristics, entity-relationship model, heuristics, normalization, view integration

## 8 Reasoning about inconsistencies in natural language requirements

Vincenzo Gervasi, Didar Zowghi

July 2005 **ACM Transactions on Software Engineering and Methodology (TOSEM)**, Volume 14 Issue 3

Full text available:  [pdf\(2.76 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)


The use of logic in identifying and analyzing inconsistency in requirements from multiple stakeholders has been found to be effective in a number of studies. Nonmonotonic logic is a theoretically well-founded formalism that is especially suited for supporting the evolution of requirements. However, direct use of logic for expressing requirements and discussing them with stakeholders poses serious usability problems, since in most cases stakeholders cannot be expected to be fluent with formal log ...

**Keywords:** Requirements, default logic, inconsistency, natural language

## 9 Query processing in deductive databases with incomplete information

Tomasz Imielinski

June 1986 **ACM SIGMOD Record , Proceedings of the 1986 ACM SIGMOD international conference on Management of data**, Volume 15 Issue 2

Full text available:  [pdf\(1.15 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We study here automated deduction in databases in the presence of various types of inference rules of the form of Horn Clauses with Skolem functions. These inference rules are typical for databases with incomplete information. We demonstrate a number of results related to processing of conjunctive queries for different types of database intensions. In particular, we show that when a database intension is built from possibly cyclic inclusion dependencies and view definitions any conjunctive ...

## 10 Technical papers: software design: Sound methods and effective tools for engineering modeling and analysis

David Coppit, Kevin J. Sullivan

May 2003 **Proceedings of the 25th International Conference on Software Engineering**

Full text available:

 [pdf\(1.33 MB\)](#)  [Publisher Site](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Modeling and analysis is indispensable in engineering. To be safe and effective, a modeling method requires

a language with a validated semantics; feature-rich, easy-to-use, dependable tools; and low engineering costs. Today we lack adequate means to develop such methods. We present a partial solution combining two techniques: formal methods for language design, and package-oriented programming for function and usability at low cost. We have evaluated the approach in an end-to-end experiment. We ...

# **11 Building knowledge base management systems**

John Mylopoulos, Vinay Chaudhri, Dimitris Plexousakis, Adel Shrufi, Thodoros Topologlou

December 1996 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 5 Issue 4

Full text available:  pdf(403.22 KB)

Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

Advanced applications in fields such as CAD, software engineering, real-time process control, corporate repositories and digital libraries require the construction, efficient access and management of large, shared knowledge bases. Such knowledge bases cannot be built using existing tools such as expert system shells, because these do not scale up, nor can they be built in terms of existing database technology, because such technology does not support the rich representational structure and infer ...

**Keywords:** Concurrency control, Constraint enforcement, Knowledge base management systems, Rule management, Storage management

# **12 Programming in an Interactive Environment: the "Lisp" Experience**

Erik Sandewall

January 1978 **ACM Computing Surveys (CSUR)**, Volume 10 Issue 1


Full text available:  pdf(3.25 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

# **13 DAIDA: an environment for evolving information systems**

M. Jarke, J. Mylopoulos, J. W. Schmidt, Y. Vassiliou

January 1992 **ACM Transactions on Information Systems (TOIS)**, Volume 10 Issue 1

Full text available:  pdf(3.63 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

We present a framework for the development of information systems based on the premise that the knowledge that influences the development process needs to somehow be captured, represented, and managed if the development process is to be rationalized. Experiences with a prototype environment developed in ESPRIT project DAIDA demonstrate the approach. The project has implemented an environment based on state-of-the-art languages for requirements modeling, design and implementation of informat ...

**Keywords:** knowledge engineering, mapping assistant, multi-level specification, repository, software information system, software process model

# **14 Generation and Verification of Finite Models and Counterexamples Using an Automated Theorem Prover Answering Two Open Questions**

Steve Winker

April 1982 **Journal of the ACM (JACM)**, Volume 29 Issue 2


Full text available:  pdf(741.49 KB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

# **15 Special issue: AI in engineering**

D. Sriram, R. Joobhani

January 1985 **ACM SIGART Bulletin**, Issue 91

Full text available:  pdf(8.79 MB)

Additional Information: [full citation](#), [abstract](#)

The papers in this special issue were compiled from responses to the announcement in the July 1984 issue of the SIGART newsletter and notices posted over the ARPAnet. The interest being shown in this area is reflected in the sixty papers received from over six countries. About half the papers were received over the computer network.

# **16 Strategic directions in software engineering and programming languages**

Carl Gunter, John Mitchell, David Notkin

December 1996 **ACM Computing Surveys (CSUR)**, Volume 28 Issue 4

Full text available:  pdf(200.92 KB)


Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**Coming to grips with a RISC: a report of the progress of the LOW RISC design group**

Jonathan W. Mills

March 1987 **ACM SIGARCH Computer Architecture News**, Volume 15 Issue 1Full text available:  [pdf\(713.42 KB\)](#)Additional Information: [full citation](#), [index terms](#)**18 NRL invitational workshop on testing and proving: two approaches to assurance**

Carl E. Landwehr, Susan L. Gerhart, John McLean, Donald I. Good, Nancy Leveson

October 1986 **ACM SIGSOFT Software Engineering Notes**, Volume 11 Issue 5Full text available:  [pdf\(1.45 MB\)](#)Additional Information: [full citation](#), [index terms](#)**19 Session 1: Mechanisms for a spatially distributed market**

Moshe Babaioff, Noam Nisan, Elan Pavlov

May 2004 **Proceedings of the 5th ACM conference on Electronic commerce**Full text available:  [pdf\(168.56 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

We consider the problem of a spatially distributed market with strategic agents. In this problem a single good is traded in a set of independent markets, where shipment between markets is possible but incurs a cost. The problem has previously been studied in the non-strategic case, in which it can be analyzed and solved as a min-cost-flow problem. We consider the case where buyers and sellers are strategic. Our first result gives a double characterization of the VCG prices, first as distances in ac ...

**Keywords:** algorithms, economics**20 Formal verification in hardware design: a survey**

Christoph Kern, Mark R. Greenstreet

April 1999 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 4 Issue 2Full text available:  [pdf\(411.53 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In recent years, formal methods have emerged as an alternative approach to ensuring the quality and correctness of hardware designs, overcoming some of the limitations of traditional validation techniques such as simulation and testing. There are two main aspects to the application of formal methods in a design process: the formal framework used to specify desired properties of a design and the verification techniques and tools used to reason about the relationship between a spec ...

**Keywords:** case studies, formal methods, formal verification, hardware verification, language containment, model checking, survey, theorem proving

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)